Method and apparatus for processing free-format data

Publication number: CN1315020
Publication date: 2001-09-26
Inventor: HETHERINGTON GREG (AU)
Applicant: GREG HETHERINGTON (AU)
Classification:
- international: ***G06F17/21; G06F17/22; G06F17/27; G06F17/30;*** **G06F17/21; G06F17/22; G06F17/27; G06F17/30;** (IPC1-7): G06F17/30; G06F17/20
- european: G06F17/21F8; G06F17/22; G06F17/22F; G06F17/22M; G06F17/22T; G06F17/27A; G06F17/27R; G06F17/27S; G06F17/30T
Application number: CN19988014202 19980422
Priority number(s): AU1997PP00439 19970422

Also published as:

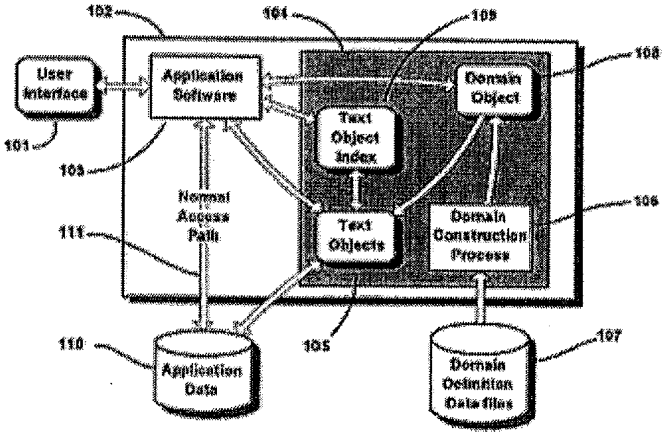
WO9848360 (A1)
EP1078323 (A1)
US6272495 (B1)
EP1078323 (A0)
CA2329345 (A1)

more >>

Report a data error here

Abstract not available for CN1315020
Abstract of corresponding document: **US6272495**

A method and apparatus for processing free-format data (301) to produce a "text object" associated with the free-format data. The text object comprises a plurality of "component nodes" (302-312) containing attribute-type identifiers for elements of the free-format text and other data facilitating access to the text object to obtain information and/or change or add the free-format data. This arrangement obviates the need for the provision of separate database fields for each element of the information. Free-format data can therefore be processed in a similar manner to the way a human being processes free-format data. All elements can be accessed via the constructed text object.



Data supplied from the ***esp@cenet*** database - Worldwide

[19]中华人民共和国国家知识产权局

[51]Int. Cl[7]
G06F 17/30
G06F 17/20

[12]发明专利申请公开说明书

[21]申请号 98814202.3

[43]公开日 2001年9月26日

[11]公开号 CN 1315020A

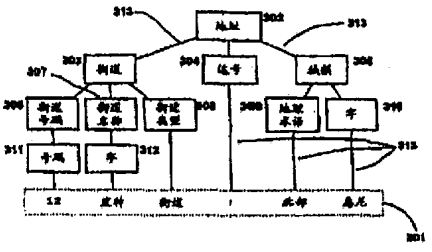
[22]申请日 1998.4.22 [21]申请号 98814202.3
[86]国际申请 PCT/AU98/00288 1998.4.22
[87]国际公布 WO98/48360 英 1998.10.29
[85]进入国家阶段日期 2000.12.15
[71]申请人 格雷格·赫瑟林顿
地址 澳大利亚新南威尔士
[72]发明人 格雷格·赫瑟林顿
[74]专利代理机构 柳沈知识产权律师事务所
代理人 马 莹

权利要求书7页 说明书48页 附图页数20页

[54]发明名称 自由格式数据处理的方法和设备

[57]摘要

本发明提供了一种处理自由格式数据(301)的方法和设备,以生成与自由 格式数据相关联的"文本对象"。文本对象包括多个"元素节点"(302—312),这些节点包括自由格式文本元素的属性类型标识符,和帮助对文本对象的访 问、取得信息和/或修改或添加自由格式数据的其它数据。这样的配置消除了 对信息的每一个元素构建独立的数据库字段的要求,这样,自由格式数据可按照人性化处理自由格式数据的方式进行处理。通过构建的文本对象可以访 问所有的元素。



ISSN1008-4274

00.12.15

权 利 要 求 书

1.一种处理存储在计算系统中的自由格式数据的方法，它包括：对数据元素的检查以确定数据的属性的步骤，通过检查元素的内容和元素之间的上下文关系来确定数据的语义和造句信息（属性），生成与这些信息相关联的附加数据，并以文本对象的形式表现出来，文本对象包括：指针装置，用它可对自由格式数据的元素进行访问，并且附加数据可以通过查询处理装置进行访问，用来向与数据的语义和造句有关的查询提供回答，和/或访问数据并进行处理。

2.如权利要求1所述的方法，其中自由格式数据作为一个记录存储在数据库自由格式字段中。

3.如权利要求1或2所述的方法，其中，数据在计算系统中按其原样进行存储，在此，它们可被其他的应用程序进行访问。

4.如前述任何一项权利要求所述的方法，其中，文本对象包括：一个属性类型标识符，用来表示数据元素的属性类型。

5.如前述任何一项权利要求所述的方法，其中，文本对象包括一个标识数据元素字符长度的值。

6.如权利要求4或5所述的方法，其中，文本对象包括一个这样的值，它用来表示一个元素是其造句层次中的一个低等级的元素还是高等级的元素，当与按该方法处理的其它数据进行匹配数据时，该值用于匹配目的。

7.如前述任何一项权利要求所述的方法，其中，文本对象包括一个数据元素的匹配程度值，它可被用来确定与其它的自由格式数据进行匹配时元素的重要性。

8.如前述任何一项权利要求所述的方法，其中，文本对象包括多个按自由格式数据的语义结构安排的元素节点，这些元素节点按自由格式数据的语义结构相应的层次排列，并且每一个元素节点包括与自由格式数据的相应元素相关的附加数据。

9.如前述任何一项权利要求所述的方法，包括进一步的步骤：生

成匹配值，用于将根据本发明方法处理的其他的一自由格式数据元素与一自由格式数据元素的比较。

10.如权利要求9所述的方法，其中的匹配值是一个语音值，用来语音比较自由格式数据元素。

11.如前述任何一项权利要求所述的方法，其中，文本对象包括与从自由格式数据中暗示得到的信息相匹配的暗示数据。

12.如前述任何一项权利要求所述的方法，其中，多个自由格式数据记录被处理，并且生成一个与每个自由格式数据记录相关联的文本对象。

13.如权利要求12所述的方法，其中，文本对象被存储在计算机系统中，可用于通过查询处理装置对于有关自由格式数据记录的查询。

14.如权利要求12所述的方法，它包括进一步的步骤：生成文本对象索引，文本对象索引包括：对于每一个数据记录的元素的属性类型标识符，以及指向每一数据记录的指针，从而，可按数据的语义和造句信息对索引进行查询，或通过索引对数据进行访问。

15.如权利要求14所述的方法，其中，文本对象索引的每一个条目，包括一个代表值关键码，它给出一个值，表示与属性类型标识符关联的元素的特点。

16.如前述任何一项权利要求所述的方法，其中，包括进一步的步骤：进行域构建，以便从域定义数据文件中构建域对象，域对象可根据语法规则通过解析自由格式数据进行检查过程。

17.如权利要求16所述的方法，其中，域定义数据文件包括：字符定义数据、正规表达定义数据和语法数据。

18.如前述任何一项权利要求所述的方法，其中，自由格式数据为邮政地址数据。

19.如前述任何一项权利要求所述的方法，其中，查询处理装置可以通过附加数据对数据进行正常的数据库操作。

20.一种处理存储在计算系统中的自由格式数据的处理系统，其设备包括：通过检查数据的元素来确定数据的属性的装置，通过检查元素的内容和元素之间的上下文关系，来确定关于数据的语义和造词信

息（属性），以文本对象的形式，生成与这些信息关联的附加数据的装置，文本对象包括：指针装置，可用来对自由格式数据的元素进行访问；以及查询处理装置，它可用来访问附加数据以便提供与数据的语义和造词信息有关的查询的答案，或/和访问数据并进行处理。

21.如权利要求20所述的处理系统，其中，自由格式数据在数据库的自由格式字段作为一个记录被存储的。

22.如权利要求20或21所述的处理系统，其中，检查装置不影响数据的存储。

23.如权利要求20-22任何一项所述的处理系统，其中，文本对象包括一个属性类型标识符用来表示数据元素的属性类型。

24.如权利要求20-23任何一项所述的处理系统，其中，文本对象包括一个说明数据元素字符长度的值。

25.如权利要求23或24所述的处理系统，其中，文本对象包括一个值，指示元素的属性类型是在造句层次中的低等级还是高等级，从而当与按照该系统处理的其它自由格式数据比较时，该值用于匹配目的。

26.如权利要求20-25任何一项所述的处理系统，其中，文本对象包括一个数据元素的匹配程度值，它可用来确定该元素与其它的自由格式数据匹配时该元素的重要性。

27.如权利要求20-26任何一项所述的处理系统，其中，文本对象包括多个根据自由格式数据语音结构排列的元素节点，这些元素节点根据自由格式数据的语音结构按层次排列，并且每一元素节点包含与自由格式数据相应的元素相关联的附加数据。

28.如权利要求20-27任何一项所述的处理系统，文本对象含义为，生成匹配值，用来进行由该处理系统处理的其他自由格式数据的一元素与该自由格式数据的一元素之间的比较。

29.如权利要求28所述的处理系统，其中，匹配值是语音值，用于语音比较自由格式数据。

30.如权利要求20-29任何一项所述的处理系统，其中，文本对象包括与自由格式数据所含的信息相关联的暗示的数据。

31.如权利要求20-30任何一项所述的处理系统，其中，该系统用来进行多个自由格式数据记录的处理，并且生成与每一个自由格式数据记录相关联的文本对象。

32.如权利要求31所述的处理系统，其中，设置生成附加数据的装置用来生成文本对象索引，该文本对象索引包括：每一数据记录元素的属性类型标识符，和指向每一数据记录的指针，并且其中，查询处理装置用来访问文本对象索引，以便向与数据的语义和造句信息相关联的查询提供回答，和/或访问数据并进行处理。

33.如权利要求32所述的处理系统，其中，文本对象索引包括各条目的代表值关键码，它给出与该条目的属性类型标识符相关联的元素的特性的表示值，以便与按本系统处理的其他的自由格式数据进行匹配。

34.如权利要求20-33任何一项所述的处理系统，进一步包括域对象，它用来通过按照语法规则解析自由格式数据进行检查处理。

35.如权利要求34所述的处理系统，其中，域对象由域构建程序从域定义数据文件中生成。

36.如权利要求35所述的处理系统，进一步包括域生成器，用于进行域构建处理。

37.如权利要求35或36项所述的处理系统，其中，域定义数据文件包括：字符定义数据、正规表达定义数据和语法数据。

38.如权利要求20-37任何一项所述的处理系统，其中，自由格式数据是邮政地址数据。

39.如权利要求20-38任何一项所述的处理系统，其中，查询处理装置用来通过附加数据对数据进行传统的数据库操作。

40.一种能够访问存储在计算系统中的自由格式数据的方法，它包括多个自由格式数据记录，并且包括步骤：存储与数据的语义和造词信息（属性）相关联的附加数据用于每一数据记录，附加数据是以与每个数据记录相关的文本对象的形式表示的，文本对象包括：指针装置，用来访问每一个自由格式数据记录的元素，附加数据可由查询处理装置进行访问，以便提供与数据的语义和造词信息相关的查询的回

答，和/或访问数据并进行处理。

41.一种处理系统，能够访问存储在计算系统中的自由格式数据，包括多个自由格式数据记录，处理系统包括用于每一数据记录的与数据的语义和造词信息（属性）相关联的附加数据，并被存储及可由处理系统进行存储和访问，附加数据是与每一数据记录相关联的文本对象的形式，文本对象包括：指针装置，用来访问每一自由格式数据记录的元素，以及查询处理装置，它可对附加数据进行访问，以便提供与数据的语义和造词信息相关的查询的回答，和/或访问数据并进行处理。

42.一种能够访问存储在计算系统中的自由格式数据的方法，包括多个自由格式数据记录，包括存储与每一数据记录中的数据的语义和造词信息（属性）相关的附加数据的步骤，附加数据是以文本对象索引的形式，包括：每一数据记录元素的属性类型标识符，和指向每一数据记录的指针，文本对象索引可由查询处理装置访问，以便提供与数据的语义和造词信息相关的查询的回答，和/或访问数据并进行处理。

43.一种能够访问存储在计算系统中的自由格式数据的处理系统，包括多个自由格式数据记录，该处理系统包括用于每一数据记录的自由格式数据的语义和造词信息（属性）相关的附加数据，附加数据是以文本对象索引的形式，包括：每一个数据记录的元素的属性类型标识符，和指向每一数据记录的指针，和查询处理装置，用来访问附加数据，以便提供与数据的语义和造词信息相关的查询的回答，和/或访问数据并进行处理。

44.一种访问按照权利要求 1-19 任何一项方法处理的自由格式数据的方法，包括步骤：访问附加数据，以便提供与数据的语义和造词信息相关的查询的回答，和/或访问数据并进行处理。

45.一种访问按照权利要求 1-19 任何一项方法处理的自由格式数据的处理系统，该处理系统包括查询处理装置，用来访问附加数据，以便提供与数据的语义和造词信息相关的查询的回答，和/或访问数据并进行处理。

00.12.15

46.一种处理存储在计算系统中的自由格式数据的处理系统，该处理系统包括对数据元素的检查以确定数据的属性的装置，通过检查元素的内容和元素之间的上下文关系以确定数据的语义和造词信息（属性），以及查询处理装置，用于利用这些信息以便提供与数据的语义和造词信息相关的查询的回答，和/或访问数据并进行处理。

47.如权利要求46所述的处理系统，其中，检查装置保持自由格式数据如同存储在计算机系统中，不影响数据。

48.一种处理存储在计算系统中的自由格式数据的方法，包括检查数据元素以便确定数据的属性的步骤，通过检查元素的内容和元素之间的上下文关系以确定数据的语义和造词信息（属性），以及利用这些信息查询数据，以便提供与数据的语义和造词信息相关的查询的回答，和/或访问数据并进行处理。

49.如权利要求48所述的处理自由格式数据的方法，其中，自由格式数据不会受检查过程的影响，并以其原样在计算系统中存放。

50. 一种存储指令的计算机可读存储器，所述指令用于按照权利要求1到19任何一项所述的方法，控制计算机对计算机系统中存储的自由格式数据进行处理。

51.一种存储指令的计算机可读存储器，所述指令用于按照权利要求48所述的方法，控制计算机对计算机系统中存储的自由格式数据进行处理。

52.一种处理存储在计算机系统中的多个自由格式数据的方法，包括步骤：对每一数据记录，进行数据元素的检查，以确定数据的属性，通过检查元素的内容和元素之间的上下文关系，确定每一项记录的语义和造词信息(属性)，并且生成与每一项记录相关联的虚拟数据字段，使能够对该信息和相关元素进行访问，从而可为每一记录提供相关联的虚拟数据字段，以便访问该记录的语义和造词信息以及也对关联元素访问。

53.一种处理存储在计算系统中的自由格式数据记录的处理系统，包括对每一记录的数据元素检查，以确定数据的属性的装置，通过检查元素的内容和元素之间的上下文关系以便确定数据的语义和造词信

息（属性），以及生成与每一记录相关联的虚拟字段的装置，能够对这些信息和相关联的元素进行访问，由此可为每一记录提供相关联的虚拟数据字段，以便对该记录的语义和造词信息进行访问以及也对相关联的元素访问。

00.12.15

说　明　书

自由格式数据处理的方法和设备

本发明一般涉及自由格式数据形式的信息的处理、存储和分析，特别涉及(但不仅限于)用于解译自由格式文本的方法和设备。

发明背景

对于计算机系统来说，其主要的目的之一就是对信息进行管理。这种信息管理是通过计算机内部的数据管理系统来实现的。

总的来说，数据管理系统可分为两大类：1）数据库管理系统；和2）文本搜索与检索系统。

第一类的数据管理系统将数据以特定的格式存入计算机内部，以便这些数据得以保存下来和进一步的编辑。在需要的时候，这些系统将对数据以特定的格式进行表达以便人类读取数据或由其他的系统使用数据。这一类的数据管理系统包括：分级、网络、关联、对象数据库系统以及智能型管理系统。

在分级网络关联数据库中，关于一个实体的信息（一项交易、库存项目、一个人、一个公司、一个地址等）通常被叫做一项“记录”（尽管有时候，一项记录可能会包括多个实体的信息），在每一项记录中，实体的各个“属性”通常又被分成“字段”。

对于对象数据库管理系统和智能型管理系统来说，这些基本的单元可能会有不同的叫法，比如“对象”，并且关于对象的信息可能被叫做“槽(slot)”或“成分”。每一个属性字段/槽均有一种格式，可能是，例如，整数、实数、布尔逻辑或字符等。其他为记录/对象。某些字段/槽具有特定的格式（例如，日期、时间），然而，还有一些则为自由格式文本。

数据库一旦建立后，可用来实现以下的操作：

- 添加一项记录/对象
- 放置和修改一项记录/对象
- 放置和删除一项记录/对象
- 检索信息

这些操作将被称为“正常的数据库操作”。

00.12.15

在字段/槽中，对于实体信息的存储适用于多种类型数据。然而，还有某些类的数据不存在适用的标准结构。没有标准结构数据的一个最好的例子便是“地址”数据。由于大多数的数据库是在一个、两个或三个自由格式字段存储人的地址信息的，按地址的单个属性来实现基本的数据库操作非常困难。注意，“属性”这个术语在这里指的是数据某个“元素”的特性。

例如，自由格式数据“悉尼北部皮特街35号”就有多个“元素”。每一个元素有一个关联的“属性”。“北部”这一元素的“属性”为“地理指示”。元素“12”的“属性”为一个“号码(number)”。值得注意的是，“低等级”的元素与数据的“令牌(tokens)”相对应，也就是说，元素“北部”是数据的一个“令牌”。然而，数据还包含高等级元素，例如，“悉尼北部”就是一个包含两个令牌的元素，这一元素的属性为“城镇”。整个数据“悉尼北部皮特街12号”的属性，也就是总的“元素”为一个“地址”。对于元素的另一个叫法为“组件”。

对于这种自由格式数据的每一个元素为其关联的属性设有其自己的字段将会大大的增大数据库的大小和复杂性，即使对于这个简单的地址实例亦是如此。对于那些包括人员以及其地址信息的数据库，为了避免过于复杂，特别是对于老的数据库，地址数据可存储在一个标有“地址”的单个字段中。在这一字段中包含自由格式的地址，并且，对于当前的数据库技术来说，对地址的各个元素实现基本的数据库操作是不可能的－那些元素是不可以单独进行访问的（那些构成地址的元素集合在一块，作为一个“地址”整体可被访问的情况除外）。

在一定程度上，这一问题受到数据库清扫/清洁科学界的重视。这一商业化奋斗的领域将解析过程应用到自由格式文本，以便生成自由格式文本属性的新的数据库字段并且跨入完全标准化数据的区域。这一数据的标准化过程包括将所有的拼写变量转化成一个一致的设定。（例如，“街道（street）→“st”）。该实例将会生成以下的结果：

房号	街道名称	街道类型	城市
12	皮特	街道	悉尼

从此，这些新的数据库字段将被用来进行基本的数据库操作。整个工业已完全的投入到这一领域，使用庞大的、复杂和贵重的软件包提取存储在数据库中的信息，分析和处理这些信息以为这些信息记录属性生成包括更多字段的新的数据库，从此为这些记录的操作提供更强的灵活性。

关于清洁/清扫数据库领域，已有很多著作（参阅，例如，1996年9月，DBMS杂志上的“处理糟杂数据”一文）。这一过程十分昂贵－对于一个庞大的数据库进行清洁操作会花费成千上百万的费用，原因在于这一过程消耗时间非常多，并且所开发的用于清洁数据库的软件包也非常的复杂——其实，即使这样，在基本的操作方面，即，就某个元素进行数据库操作上仍存在着局限性，一个元素必须有一个自己的字段。

这就将我们带入到第二个主要的问题，这一问题一直困扰着如何在商业化数据库中存储计算机化的信息。事实上，所有的商业数据均存储在分级式的、关联式的数据库或平面数据文件中，这些文件具有在设计时就已确定的具体的结构，然而，从这些信息的本质上看将是非常复杂的，并且甚至具有无限数量的不同属性。要生成一个用于每一个和所有的不同类型的信息的每一和所有属性的字段的数据库，事实上是不实际的。假定是有可能的话，要生成一个包含用于人类所需的所有类型信息的字段的数据库，从成本上讲，也是不实际的。

既使一个费用不大的（但很重要的）例子就足以说明问题的难度。让我们考虑一下国际地址问题，也就是，世界上所有的地址。尽管四或五个自由格式字段就可以包含任何地址，要设计一个包括所有的国际地址的所有可能的属性的数据字段这样的一个数据库将需要不是上千个也会是上百个这样的数据字段。英国有多个县；美国和澳大利亚有多个州；日本有多个区，并且各个不同的地址又有各自不同的表达习惯等等。

数据库清洁/清扫的研究是迄今为止仅有的部分方案。然而，对于每一数据属性，它仍然需要相同的基本的数据库结构。人们可以建立越来越多的复杂的数据库，但是这一问题终不会彻底的解决，并且会大大地限制计算机对信息的处理。

自然语言处理系统是通过使用“语义语法”对语义信息编码使其成为一种依据造句法的语法。这些系统主要用来向其他的系统，例如数据库管理系

统，提供自然语言接口。以下的内容摘自由D.W.皮特森(Patterson, D.W.)写的一本书“人工智能和专家系统”。

“...它们采用无终端的语义成分使用与上下文无关的改写规则。这些成分为类别或元符号，例如，属性、对象、呈现（如出现在显示器上的或打印机上的）、和船舶(ship)，而不是NP（名词状态）、VP（动词状态）、N（名词）、V（动词）等等。...在为数不多的应用中，语义语法已被证明是成功的，这些应用包括LIFER，有（美国）海军发布的一个数据库查询系统...，和用于指导电路故障诊断的，名为SOPHIE的指南系统。

从根本上讲，用于这些系统中的改写规则使用以下的格式

S → 什么是<CIRCUIT-PART>的<OUTPUT-PROPERTY>?

OUTPUT-PROPERTY → 该<OUTPUT-PROP>

OUTPUT-PROPERTY → <OUTPUT-PROP>

CIRCUIT-PART → C23

CIRCUIT-PART → D12

OUTPUT-PROP → 电压

OUTPUT-PROP → 电流

在LIFER系统中，有处理大量的查询问题的格式，例如

最靠近纽约的通信公司(carriers)的名字是什么?

谁指挥肯尼迪?

等等。

分析这些句子并把单词对照为辞典词条中的元符号。例如，输入句子“打印计划的长度” ('print the length of the Enterprise')将与

<LTG> -> <PRESENT> the <ATTRIBUTE> of <SHIP>相配

其中，print与<PRESENT>相配；length与<ATTRIBUTE>相配；Enterprise与<SHIP>的LIFER顶层语法规则相配。与<ATTRIBUTE>相配的其他典型的辞典词条包括等级、指挥官、燃料、类型、梁和长度等等（CLASS, COMMANDER, FUEL, TYPE, BEAM, LENGTH）。

这些类型的系统以特定结构的或自由格式的形式接收数据并将其转化为系统自己的表达形式。

尽管其接口具有一定的灵活性，并且与之连接的数据库具有固定的结构形式，然而，这些系统仍无法对数据的原文（人类可读的）进行转化。

的确，以前是有多个系统，它可向具有特定结构的数据库提供“自然语言”的接口。不过，所有的这类系统均是将“自然语言”翻译成某些特定结构数据的格式。这同样存在前面所述的同样的问题。

作为这类系统的一个实例，请参阅美国专利4787035，波恩，“数据译员”和美国专利5454106，波斯，玛哈托，A.，“使用自然语言展示...可读元素的数据库检索系统”。

正如前面所讨论的，有一种数据库管理系统是智能型的数据库管理系统（KBMS）。

这些系统对对象采用属性“槽”的概念。这些槽就对象提供或是转换信息，进行直接的存储或是间接的经过处理。一个简单的“槽”的实例就可以说明问题：一个“正方形”对象有两个属性槽，“长度”和“面积”。“面积”槽的值不需要进行存储，因为它可以通过“长度”值的平方而求得。

尽管这类的系统不需要固定的数据库结构，然而，它们却需要将原始数据转化成内部数据表示方法，这就需要一套非常复杂的“语言生成”过程以便生成人类可读的信息。如果要求这类的系统保留原始数据以便其他的系统或人类使用，一个小小的转化就需要整个的文本串重新生成。

数据管理系统的文本搜索和检索类别不是输入数据，而是建立可搜索的指向原始数据的索引。这些类别包括：文件存储与检索系统；以及互联网搜索引擎。

这类的系统之所以非常成功，原因在于它们将原始信息保留成人类可读的形式。这一基本的原则意味着，它不同于前面提到的原始的数据库系统，其根本的数据可以轻松的与该类很多的系统共享。其成功的另一个原因在于，在不需要对原始数据进行转化的情况下即可进行技术方面的革新。数据转换不仅十分昂贵，而且也是数据出错的主要原因。

然而，与上面所述的数据库系统相比，在使用这类系统管理数据时，它也存在着诸多的缺点。主要的缺陷是数据无法操纵-它不可修改，什么样就是什么样。难于进行操作的数据库功能还有：

- 对数据进行交互检查和确认
- 将数据与数据库系统集成
- 对文本数据进行分类和分级

从这些局限性上我们可以看到，该类数据管理系统适合于不需要进行转换的未结构化的数据。

在文本搜索与检索系统中，众所周知，在对文件库进行处理时，需要找出每一个文件的特定的属性，比如，“主题”词。使用这样的系统进行处理的文件类型包括书、报纸、报告、手册和电子邮件。

然而，大多数这类系统只寻找单个的字进行匹配，而不是按上下文找多个字。还有一些其他的系统通过识别名词却不对名词进行分类。这两种系统均不适合诸如地址数据这样的数据，因为它包含着一大部分的专有名词。

另外，在上下文中，原始数据无法更改。

关于这一领域的详细情况，请参阅盖拉德·萨敦的著作。

请注意，不要把在下文中将使用的“文本对象”这个术语与描述在计算机系统之间通过压缩文本串对一项项的文本数据进行存储和转换的软件技术中使用的术语“文本对象”相混淆。使用“文本对象”这一术语的技术从用于苹果机操作系统的“串”对象（其中的对象主要包含两字节“长度”值的导引和文本串）到X-Windows操作系统中使用的“复合串”对象（其中的对象对一项信息的多重编码、语言转换、和字体进行压缩）均有使用。

发明概述

本发明的第一方面提供了一种对存储在计算机系统中的自由格式数据的处理方法。它包括对数据元素的内容进行检查的步骤，以便确定数据的属性，通过检查各个元素的内容以及各个元素之间的上下文关系，来确定有关数据的语义和造句信息（属性），生成与这些信息相关联的附加的数据，并以文本对象的形式表示，这包括指针装置，它意味着可对自由格式数据的元素进行访问，并且，可以通过查询处理装置访问附加数据，以便提供与数据的语义和造句信息有关的查询的答案，和/或访问数据并对数据进行操作。

在此说明书中使用的术语“文本对象”并不象前面所述的那样对文本串进行压缩。在本发明中的文本对象这一术语提供了一个实际文本数据和例如一个需要访问和/或处理文本数据的应用软件系统之间的“语义层”。

按其最简单的格式，如前面所说的，文本对象即为附加数据，与通过对数据元素的检查所得到的语义和造句信息相关联，并且指针装置为（例如一个关键码(key)），它可以带回到自由格式数据的元素（例如，查索形成自由格式数据的文本串）。

附加数据最好允许通过对数据的检查所得到的数据的属性。例如，在前言部分给出的那个例子-“悉尼北部皮特街12号”中，该数据具有各不相同的属性，例如，“街道(street)”等于“皮特街12号：”；“街道号码(street number)”等于“12”；“城镇(town)”等于“悉尼北部”，等等。这些属性由附加数据进行辨识，并且，指针装置最好允许对这些数据的元素进行访问，这些数据的元素又与那些属性相关联。附加数据有效的提供了一些“虚拟数据字段”-这些数据字段不像他们在一般的数据库中那样真实的存在，并有一栏目字段前端用于放置每一项属性。不过，基于一个一个的属性的基础上，使用本发明便可对自由格式数据进行访问，犹如那些属性字段真实地存在着一样。本发明优选的体现方法是生成“虚拟数据字段”，它最好允许对自由格式文本的一般的数据库操作，而没有必要生成自由格式文本的真实的数据字段。自由格式文本能够在其同样的位置保持其原样存储（通常为数据库）。

当有人考虑对多个自由格式数据记录进行处理时，如国际地址数据，本发明的重要性将显而易见。正如上面所讨论的，尽管四到五个地址字段便可存储自由格式形式的所有的国际地址数据，然而，每项数据记录会有多个属性，而且与其他的地址相比，其属性又各不相同，比如，英国有县，而美国有州。因此，为所有的国际地址的所有的属性生成真实的传统的数据库字段简直就是一项几乎是不可能的任务。然而，对于本发明来讲，自由格式数据的每一项记录均可以实现并且进行处理，并继而以文本对象的形式为该特定的记录生成一个数量不大的虚拟数据字段。然后，通过恰当的查询处理装置对于每一项记录的文本对象进行单独的查询，以便对于该项记录提供所有的一般数据库操作。数据本身可保留其原位。由于为每一个记录生成了一个单独的文本对象，对于每一个记录提供不同的虚拟数据字段是不成问题的。我

们没有必要生成带有多个字段的庞大的数据库，我们只需使数据库记录保留其原样并且生成多个文本对象，每个记录生成一个，总体上给出多个虚拟字段，然而，对于每个文本对象没有几个虚拟字段。

检查的步骤包括对自由格式数据的解析过程。

一个文本对象最好能够使数据处理，进行所有一般的数据库操作，例如，修改记录、存放记录的元素、从记录中检索信息等等。由文本对象所提供的信息最好包括数据元素的信息。在优选的实现方式中，信息也可能包括其匹配信息（例如语音），以帮助数据的一项记录与数据的另一项记录之间进行比较，筛选出优先的信息以更加有效地对自由格式文本等进行处理。

有理由相信，这一新的方法将使计算机能够几乎像人类那样对自由格式数据进行处理。一旦数据库的恰当的栏目名称被确定下来，就没有必要根据数据记录的属性分解数据记录，并将每一属性类型的标准值放入数据库中适当的字段中（如传统的做法那样）。每一项数据记录的每一个文本对象均提供了计算机所需的所有的处理方法和信息，以便进行一般的数据库操作。比如说，国际地址的属性类型是可以比较，处理的，等，没有必要提供一个带有多个字段的复杂的数据库。

文本对象包括可访问的属性类型标识符和指针。前者用来对自由格式数据的属性进行识别；后者用来对具有特定属性的数据元素进行定位。

在优选的实施例中，文本对象包括“元素节点”式的多个组件。更可取的是，多个元素节点按照事先确定的层次在文本对象中是相互关联的。例如，这些元素节点可被看作是以“文本节点树”的形式“筑巢”在一起，该树有许许多多的树枝，它又将多个不同的元素节点按预定的层次相互关联在一起，每一个元素节点可包括：

- 一个属性类型标识符（对与该元素节点相关联的自由格式数据的属性进行分级）；
- 一个指针，指向文本对象串内的次级串的开始（也就是说，与元素节点相关联的元素的开始点）。
- 一个整数，包含（数据）元素次级串的字符长度。
- 零、一或多个其他的元素节点（在这个元素节点内筑巢或与该元素节点关联在一起以便可通过该元素节点对其他的元素节点进行访问），最好以队列的形式存储。

- 一个匹配程度（matching weight）（在与其他的文本对象进行比较时，表示该元素的相对重要性）；
- 一个布尔逻辑变量，表示该属性类型标识符是否是一个低等级的匹配元素，和
- 取决于时间/空间上的考虑，一个或多个值，以便于匹配处理过程。（对此，请参阅下面的“文本串操作”部分）
- 一个解析优先级值（对与元素节点相关联的自由格式数据的元素给出一个概念性的优先级，以便对优先级进行分配，并在出现冲突的情况下，可使用这个优先级来确定对自由格式文本的最佳的解释）。

从形体上看，其他的元素节点可能不是在元素节点内筑巢，而是每个元素节点可能只包含一列指针指向从属的元素节点，以便从属的元素节点在包含该列指针的元素节点中被识别出来。

更为可取的是，每一元素节点与自由格式数据的一个特定的属性相关联，正如被元素节点中属性类型标识符所识别出来的那样。那些层次相对高的元素节点可包含或指向多个其他的元素节点，而那些层次最低的元素节点可不包含或指向任何其他的元素节点，因为比其层次再次级的便是自由格式数据的关联元素了。

层次是由自由格式数据的解析过程确定的。例如，一项地址数据记录的某个属性可能是<街道>，也就是说，“皮特街12号”。<街道>元素的次级属性是<街道号码>“12”、<街道名称>“皮特”和<街道类型>“街道”。<街道>元素节点将由此列出三个其他的次级元素节点，具有属性类型标识符<街道号码>、<街道名称>和<街道类型>。

更为可取的是，每一个元素节点可被看作是文本对象本身。这一递归特点使得本发明的文本对象的所有的功能可应用于每一项属性。

文本对象也可包含其他的数据结构，用于特定元素节点的快速定位。这种结构的一个实例就是一个查阅表，它含有所有的属性类型标识符和指针指向与它们相关联的元素节点。

更可取的是，查询处理装置是一套应用软件引擎，通过对其配置，它能够使用文本对象回答有关数据的问题和访问数据以便对其进行处理（例如，有错误时进行纠正）。

更可取的是，这种方法还包含准备进一步的“索引”的功能，这可更有利于对自由格式数据的多项记录的元素进行比较。更可取的是，索引是以一个表格的形式形成的（发明人称之为“文本对象索引”），它包括，栏目、栏对象题和数据。除了它是从附加数据中为多个数据记录的每一个数据记录准备的以外，就像传统的数据库的方式是一样的。

更可取的是，文本对象索引包含一个表格，表格中有一个属性类型标识符的栏目，一个代表值关键码的栏目，和一个用户提供的记录标识符的栏目。代表值关键码提供与恰当的元素类型标识符关联的元素的特性的代表值，例如，专有名词元素（例如，史密斯）的语音值或普通词（比如，街道）的数字标识符。下面的关于文本串匹配的内容更为详细的介绍有关代表值的情况。用户提供的记录标识符将为用户辨识哪一个自由格式数据记录将被比较或访问，也就是一个指针，它使得对该记录进行访问成为可能。

在准备了文本对象索引的地方，拥有多个元素节点、这些节点含有属性类型标识符和其他数据的文本对象不是十分必要。所有的访问数据和进行数据库操作所需的只是查询处理引擎和文本对象索引。文本对象索引可从对数据的检查中直接准备，并且，文本对象索引包括多项记录的文本对象（也就是说，附加数据加上指向记录的指针）。作为一个独立的“元素节点结构”，文本对象因而可被分配，或者作为一个独立的整体，不是非需要不可的。事实上，它是作为附加数据和指针与文本对象索引合成一体的。

在文本对象包含对于自由格式文本的低等级元素的“匹配”值的地方，对包含用不同的语言书写的记录的元素进行比较是完全可能的。例如，通过对其各自的匹配值进行比较，一个包含用日本汉字书写的街道名称这样的自由格式记录完全可以与用阿拉伯语书写的街道名称的自由格式记录进行比较。对于每一项记录中的街道名称可能是相同的，然而，在自由格式数据中，只是用不同的语言表达出来。由本发明的该方面所提供的这一匹配信息因而能够使得用不同语言表达的自由格式文本元素可进行比较。

匹配值可在对文本对象处理的过程中生成，并且没有必要在文本对象中存储。也就是说，它们是通过由查询处理引擎指定的程序以“飞行式(on the fly)”方式生成的。随后有详细的介绍。

在本发明的这一方法中，更可取的是，检查数据的元素以确定元素的步骤包括按照域对象的语法规则解析自由格式数据的步骤。域对象是由域构造

过程建立的。该过程使用的输入数据为：字符定义数据、正规表达定义数据和语法数据。

文本节点树的元素节点的层次优选由那个特定的域对象的语法规则来确定。

本发明的一种实现方法可由一套应用软件来实现，该软件包含一个域对象和一个查询处理装置。通过对域对象的安排，来检查自由格式数据，以便生成文本对象，然后它将被查询处理装置使用来对自由格式数据库进行操作。自由格式数据可以任何的方式进行存储，例如计算机系统上的传统的数据库。无格式数据也可作为一个数据串存储于文本目标中。包含域对象和查询处理引擎的应用软件可被用来处理数据，而不会影响数据库中的存储。因此，其他的应用软件可以通常的方式与数据库进行接口，也就是说，就其操作这一方面而言，数据库完全不受任何的影响，只是域对象和查询处理装置可被用来增强数据库的能力，因为它们提供了对自由格式数据的所有元素进行访问的能力。

没有数据清洁和准备具有更多字段的新的数据库，对自由格式数据字段的数据进行访问在以前是不可能的。除此之外，本发明还具有大大的潜力，来对数据进行将来的构建和定制。例如，利用本发明可大大的降低所需的用来在数据库中存储数据的字段的数目。就以上那个国际名称和地址的实例来看，目前不可能有一个数据库在仅仅一个字段内来处理国际地址数据。这是因为国际地址数据拥有多个不同的属性。然而，使用本发明，国际地址可储放在仅仅一个自由格式字段中，该字段包含了所有的国际地址记录。通过本发明的处理方法可向每一个国际地址记录提供其自己的一套虚拟数据字段，在其中，通过对每一数据记录所有元素信息的查询处理装置、处理和访问并与其他的记录进行比较。确实，它可提供用于所有国际地址的仅仅一个域对象。任何的自由格式数据均可以这样的方式进行处理。本发明并不仅限于地址数据。

从进一步的方面看，本发明提出了一种能够访问在计算系统中存储的自由格式数据的方法，包括多个自由格式数据记录，存储对于每一数据记录中数据的、与语义和造句信息（属性）相关联的附加数据的步骤，附加数据采用的是与每一个数据记录相关联的文本对象的形式。文本对象包括指针装置，凭借该指针能够访问每一个自由格式数据的元素。附加数据可通过查询处理

装置进行访问，以便提供与数据的语义和造词信息相关联的查询的答案，和/或访问数据并对其进行处理。

更为可取的是，文本对象包括任何或所有的文本对象的特性，正如在本发明的第一个方面中所讨论的那样，文本对象是通过检查属性特点而生成的，这已在上面进行了讨论。

本发明进一步提出了一种能够访问在计算系统中存储的自由格式数据的方法，包括多个自由格式数据记录，该方法包括存储对于每一数据记录中数据的、与语义和造句信息（属性）相关联的附加数据的步骤，附加数据采用的是文本对象索引的形式，它包括，对于每一个数据记录的元素的属性类型标识符和指针指向每一数据记录。文本对象索引可通过查询处理装置进行查询，以提供与数据的语义和造词信息相关联的问题的答案，和/或访问数据并对其进行处理。

更为可取的是，文本对象包括任何或所有的文本对象索引的特性，正如在本发明的第一个方面中所讨论的那样。文本对象是通过处理步骤而生成的，这已在上面进行了讨论。

从更进一步的方面看，本发明提供了一个处理系统，用以处理存储在计算系统中的自由格式数据。该设备包括检查数据元素以确定数据的属性的装置，通过检查元素的内容以及元素之间的上下文关系来确定有关数据的语义和造句信息（属性）；以及与该信息相关联的附加数据的生成方法，并以文本对象的形式表示出来，它包括指针装置，它能够用来访问自由格式数据的元素，和一个查询处理装置，通过该装置可对附加数据进行访问以便提供与数据的语义和造词信息相关联的查询的答案，和/或访问数据并对其进行处理。

根据任何一个或所有的特性，通过应用同样的检查方法。检查装置和生成装置用来生成一个文本对象，这在前面已进行了讨论。

本发明又进一步提供了一个处理系统，它能够对存储在计算系统中的自由格式数据进行访问，这包括多个自由格式数据记录，处理系统包括与每个数据记录的数据的语义和造句信息（属性）相关联的附加数据，这可通过处理系统进行存储和访问。附加数据以与每个数据记录相关联的文本对象的形式表示，文本对象包括：指针装置，凭此可对每个自由格式数据记录的元素

进行访问，和一个查询处理装置，用来访问附加数据，以便提供与数据的语义和造词信息相关联的问题的答案，和/或访问数据并对其进行处理。

本发明又进一步提供了一个处理系统，它能够对存储在计算系统中的自由格式数据进行访问，这包括多个自由格式数据记录，处理系统包括与每个数据记录的自由格式数据的语义和造句信息（属性）相关联的附加数据。附加数据以文本对象索引的形式表示，文本对象索引包括：每一数据记录的元素的属性类型标识符，和指针装置，指向每一数据记录，和一个查询处理装置，用来访问附加数据，以便提供与数据的语义和造词信息相关联的问题的答案，和/或访问数据并对其进行处理。

本发明又进一步提供了一种设备，包括域对象，用来处理自由格式数据从而生成文本对象，文本对象包括任何一个或所有的文本对象的特性，这在本发明的前面的几个方面中进行了说明。

在优选的实施例中，访问文本对象的过程包括对一个或多个文本对象查询其属性，并取得与查询的属性相匹配的元素的值。例如，在自由格式数据为名称和街道数据的地方，某人可能要查询其文本对象或多个对象，以便了解是否存在一个<街道>的元素，并且，如果是的话，找到该元素的值（例如，“皮特街12号”）。这一点是目前的数据库难以做到的，因为“街道”字段仅仅包括以自由格式形式的所有的<街道>。其他的一些老的系统提供的是搜索工具，用它来寻找特定的文本串，而不考虑要搜索文本的语义。这些系统只能够通过搜索该文本串用来寻找名称为“皮特”的所有的地址。当要搜索的文本可以不同的方式使用时，这就出现了问题。

检查一下在附图的图2中的地址的例子，系统用户想在这一数据内确定所有的在“包克斯路”上的地址。如果用户查找“包克斯路”，系统将返还记录201，但丢掉了记录205和207。如果用户改变对应“包克斯”的文本，系统将返还所有需要的记录，但仍错误地返还记录202、203、204和206。既使用户在每次查询中指定了“路”的每个改变，也不会得到正确的结果。如果系统用户希望考虑到数据中的错误，问题将变得更加困难，例如，当制订“包克斯路”时，返还记录206。

另一个例子是，当“街道名称”具有与“城镇名称”同样的名称时，在不考虑语义的情况下，进行串搜索时将会出现错结果，比如说，“墨尔本悉

尼大街123号”。串搜索将不会只找到只带“悉尼”的记录作为它们的城镇名称。

访问文本对象的过程也可包括对两个文本对象的比较并且认定和提供一个可信的值，该值标明了这两个文本对象相匹配的程度。例如，通过比较各自的文本对象，来对两个街道地址进行比较，取决于它们相匹配的程度，一个信任值（以百分数）便可给定。

访问过程也可包括对与某个特定的元素相关联的值的修改的过程。通常的例子有，结婚后修改一位妇女的姓，和，错误发生后，修改街道或城镇的名称。

也由许多政府改变街道，邮政编码，例如澳大利亚的北部地区修改了其邮政编码，从5800－5999变为0800－0899，甚至改变整个城市的名称的情况，例如城市列宁格勒变为皮特伯格。

本发明能够修改文本原件的特定元素的值的能力具有其优势所在，即，直接使用数据（也就是说，不使用文本对象）的传统的计算机的操作将不会受到影响。

然而，从再进一步的方面看，本发明提供了一个处理系统，它能够访问根据权利要求1-19中的任何一种方法处理的自由格式数据，该处理系统包括查询处理装置，用来访问附加数据，并提供与数据的语义和造词信息相关联的问题的答案，和/或访问数据并对其进行处理。

检查装置包括按照以上给出的任何或所有的方法步骤访问文本对象。

本发明然而又进一步提供了一种处理系统，用于处理存储在计算系统中的自由格式数据，这包括，检查数据的元素的装置，以便确定数据的属性，通过检查元素的内容和元素相互之间的上下文关系，来确定数据的语义和造句信息（属性），和一个查询处理装置，以便利用这些信息提供与数据的语义和造词信息相关联的问题的答案，和/或访问数据并对其进行处理。

检查装置包括一个域对象，它用来检查元素，并提供虚拟数据（这些数据是一些与数据的语义和造句信息相关联的数据），这些数据将被查询处理装置使用，来访问数据并得到有关数据属性的信息。

本发明然而又进一步提供了一种处理装置，用于处理存储在计算系统中的自由格式数据，这包括，检查数据的元素的步骤，以便确定数据的属性，通过检查元素的内容和元素相互之间的上下文关系，来确定数据的语义和造

句信息（属性），并利用这些信息查询这些数据提供与数据的语义和造词信息相关联的查询的答案，和/或访问数据并对其进行处理。

再进一步，本发明提供了一种处理多个存储在计算系统中的自由格式数据记录的方法，对于每一项记录检查其数据的元素以确定数据的属性，通过检查元素的内容和元素相互之间的上下文关系来确定数据的语义和造词信息（属性），并且生成虚拟数据字段，能够对这些信息进行访问，并且生成了与每一数据记录匹配的元素，在此，每一项记录均被提供了匹配的虚拟数据字段，从而能够访问语义和造词信息，以及访问相关联的元素。

在此，“虚拟数据字段”这个术语与前面提到的具有同样的含义。不同于以前的传统的数据库，它们需要处理信息并生成真实的数据字段，这里，没有必要生成独立的数据字段。数据可在数据库中保持其原样，而是为其语义和造词信息属性生成一个相关联的“虚拟字段”，并可对虚拟字段进行查询，而得到所需的记录的所有信息，并且，最好是，所有正常的数据库操作可以被实现。

本发明又进一步提供了一种处理系统，用于处理多个存储在计算系统中的自由格式数据记录，它包括，对每个记录的数据的元素检查以确定数据的属性的装置，通过检查元素的内容和元素相互之间的语义关系，来确定每项记录的语义和造词信息（属性），和生成与每一个记录相关联的虚拟数据字段以便能够对这些信息以及与其相关联的元素进行访问的装置，在此，为每一项记录提供了相关的虚拟数据字段，以便能够对该记录的语义和造词信息以及相关联的元素进行访问。

优选实施例的详细描述

参照附图，通过实例的方式，对本发明的实施例进行说明，其特点和优势会变得显而易见。其中：

图1是根据本发明的实施例对自由格式数据进行处理的系统的结构；

图2说明了样例“地址”数据；

图3是一个更为详细的自由格式文本实例的结构图，是由本发明对自由格式数据操作实施例生成的；

图4说明了样例“地址”格式；